IN THE CLAIMS

Please amend the claims as follows.

1. (Currently Amended) A method for performing multiplication on a field programmable gate array, comprising:

generating a product by multiplying a first plurality of bits from a first number and a first plurality of bits from a second number using a digital signal processor (DSP) configured to perform multiplication on a fewer number of bits then those forming the first and second numbers;

retrieving a stored value designated as a product of a second plurality of bits from the first number and a second plurality of bits from the second number from a memory;

scaling the product with respect to a position of the first plurality of bits from the first number and a position of the first plurality of bits from the second number and scaling the stored value with respect to a position of the second plurality of bits from the ~~second~~first number and a position of the second plurality of bits from the second number; and

summing a scaled product and a scaled stored value to generate a value representing a product of the first number and the second number, wherein the first number and the second number each have a number of bits equal to or greater than a total of the first and second plurality of bits.

2. (Canceled)

3. (Original) The method of Claim 2, wherein the DSP block is configured to multiply two numbers of equal bit length.

4. (Canceled)

5. (Original) The method of Claim 1, wherein scaling the product comprises shifting bits in the product relative to a global least significant bit.

6. (Original) The method of Claim 1, wherein scaling the stored value comprises shifting bits in the product relative to a global least significant bit.

7. (Original) The method of Claim 1, further comprising:

retrieving a second stored value designated as a product of a third plurality of bits from the first number and a third plurality of bits from the second number;

retrieving a third stored value designated as a product of a fourth plurality of bits from the first number and a fourth plurality of bits from the second number;

scaling the second stored value with respect to a position of the third plurality of bits from the first number and a position of the third plurality of bits from the second number and scaling the third stored value with respect to a position of the fourth plurality of bits from the first number and a position of the fourth plurality of bits from the second number; and

summing a scaled second stored value and a scaled third stored value.

8. (Original) The method of Claim 7, further comprising:

scaling a sum of the scaled product and the scaled first stored value;

scaling a sum of the scaled second stored value and the scaled third stored value; and

summing a scaled sum of the scaled product and the scaled first stored value and a scaled sum of the scaled second stored value and the scaled third stored value.

9. (Original) The method of Claim 1, further comprising:

retrieving a second stored value designated as a product of a third plurality of bits from the first number and a third plurality of bits from the second number;

retrieving a shifted value designated as a product of a fourth plurality of bits from the first number and a fourth bit from the second number;

scaling the second stored value with respect to a position of the third plurality of bits from the first number and a position of the third plurality of bits from the second number and scaling the shifted value with respect to a position of the fourth plurality of bits from the first number and a position of the fourth bit from the second number; and

summing a scaled second stored value and a scaled shifted value.

10. (Original) The method of Claim 9, further comprising:

scaling a sum of the scaled product and the scaled first stored value;

scaling a sum of the scaled second stored value and the scaled shifted value; and

summing a scaled sum of the scaled product and the scaled first stored value and a scaled sum of the scaled second stored value and the scaled shifted value.

11. (Currently Amended) A method for implementing a multiplier on a field programmable gate array, comprising:

configuring a digital signal processor (DSP) to perform multiplication on a first plurality of bits from a first number and a first plurality of bits from a second number;

storing products resulting from multiplication of a second plurality of bits from the first number and a second plurality of bits from the second number in a memory;

routing an output from the DSP to an adder such that the output from the DSP is scaled according to a position of the first plurality of bits from the first number and a position of the first plurality of bits from the second number;

routing an output of the memory to the adder such that the output from the memory is scaled according to a position of the second plurality of bits from the first number and a position of the second plurality of bits from the second number; and

outputting a value representing a product of the first and second number where the first and second number each have more than the first plurality of bits, wherein the DSP is configured to support multiplication of no more than the first plurality of bits.

12. (Original) The method of Claim 11, further comprising:

storing products resulting from multiplication of a third plurality of bits from the first number and a third plurality of bits from the second number in a second memory;

storing products resulting from multiplication of a fourth plurality of bits from the first number and a fourth plurality of bits from the second number in a third memory;

routing an output from the second memory to the adder such that the output from the second memory is scaled according to a position of the third plurality of bits from the first number and a position of the third plurality of bits from the second number; and

routing an output of the third memory to the adder such that the output from the memory is scaled according to a position of the fourth plurality of bits from the first number and a position of the fourth plurality of bits from the second number.

13. (Original) The method of Claim 11, wherein configuring the DSP comprises determining a number of bits that the DSP will multiply.

14. (Original) The method of Claim 11, further comprising determining a number of the second plurality of bits from the first number and a number of the second plurality of bits from the second number.

15. (Original) The method of Claim 11, wherein routing the output from the DSP has the effect of shifting the output from the DSP to a more significant bit position.

16. (Original) The method of Claim 11, wherein routing the output from the memory has the effect of shifting the output from the memory to a more significant bit position.

17. (Currently Amended) A multiplier, comprising:

a digital signal processor (DSP) configured to perform multiplication on a first plurality of bits from a first number and a first plurality of bits from a second number;

a memory that stores products resulting from multiplication of a second plurality of bits from the first number and a second plurality of bits from the second number; and

an adder that sums a scaled output of the DSP and a scaled output of the memory to output a value representing a product of the first and second number where the first and second number each have more than the first plurality of bits, wherein the DSP is only configurable to support multiplication of a number of bits equal to or less than the first plurality of bits.

18. (Original) The multiplier of Claim 17, wherein the DSP, the memory, and the adder reside on a field programmable gate array.

19. (Original) The multiplier of Claim 17, further comprising a second memory that stores products resulting from multiplication of a third plurality of bits from the first number and a third plurality of bits from the second number.

20. (Original) The multiplier of Claim 19, wherein the adder sums a scaled output of the second memory with the scaled output of the DSP and the scaled output of the memory.

21. (New) A method for implementing a multiplier on a field programmable gate array, comprising:

configuring a digital signal processor (DSP) to perform multiplication on a first n bits from a first number and a first n bits from a second number;

storing products resulting from multiplication of a second m bits from the first number and a second m bits from the second number in a memory;

routing an output from the DSP to an adder such that the output from the DSP is scaled according to a position of the first n bits from the first number and a position of the first n bits from the second number;

routing an output of the memory to the adder such that the output from the memory is scaled according to a position of the second m bits from the first number and a position of the second m bits from the second number; and

outputting a value representing a product of the first and second number where the first and second number each have at least n + m number of bits.

22. (New) A multiplier, comprising:

a digital signal processor (DSP) configured to perform n*n multiplication on a first plurality of n bits from a first number and a first plurality of n bits from a second number;

a memory that stores products resulting from multiplication of a second plurality of bits from the first number and a second plurality of bits from the second number; and

an adder that sums a scaled output of the DSP and a scaled output of the memory to output a value representing a product of the first and second number where the first and second number each have more than n bits.